IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-12, Dec 2018 E-ISSN: 2347-2693

Concurrency control by Multiple Granularity of Locks in Multiusers Database Environment

Anil Kumar Singh

Department of Information Technology, Jagran Institute of Management, Kanpur -208 014

*Corresponding Author: anil.sysadmin@gmail.com

Available online at: www.ijcseonline.org

Accepted: 23/Dec/2018, Published: 31/Dec/2018

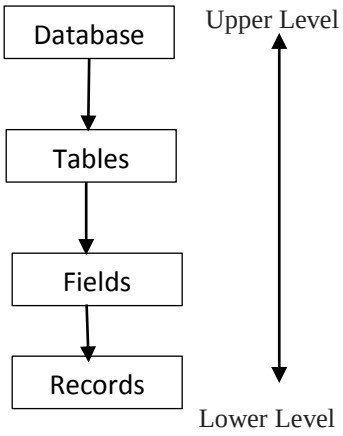
***Abstract*-**Concurrency control is a big problem in shared database system. The management of the concurrent execution of transactions in a multiuser database system is recognized as concurrency control. The aim of concurrency control is to confirm the serializability of transactions in a multiuser database situation. It is significant because the simultaneous execution of transactions over a shared database can create several data integrity and consistency problems i.e. problem of summarization and problem of maintaining statistics. Multiple granularity means which level lock will apply, if we apply the lock on exact level of database, then definitely we may control the concurrency. In this paper we shall reveal in which level lock should apply for prevent above mentioned problems.

With the help of Multiple Granularity, we shall reveal how to prevent the problem arise due to concurrency with the help of suitable examples. It will help the students and research scholars to understand that how to prevent the concurrency problems with the help of Multiple Granularity Locking Protocol method.

***Keywords*:** Multiple Granularity, Implicit Lock, Explicit Lock, Concurrency, Tuples, Database, Table, Record, Attribute

I. INTRODUCTION

A vital property of transactions is that they are lonely. Technically, this means that the execution of transactions has the same outcome as running the transactions serially, one after the next, in sequence, with no intersection in executing any two of them. Such executions are called serializable, meaning "has the same effect as a serial execution." [1] Granularity means the size of data item permissible to lock. Now Multiple Granularity means hierarchically breaking up the database into chunks which can be locked and can be track what essential to lock and in what manner. Such a hierarchy can be signified graphically as a tree. The hierarchy of database as follows:



For example, consider the tree, which contains of four stages of nodes. The upper level represents the entire database. Under it are nodes of type Tables; the database consists of precisely these tables. Table has children nodes which are called fields.

Finally, each field has child nodes called record [2].

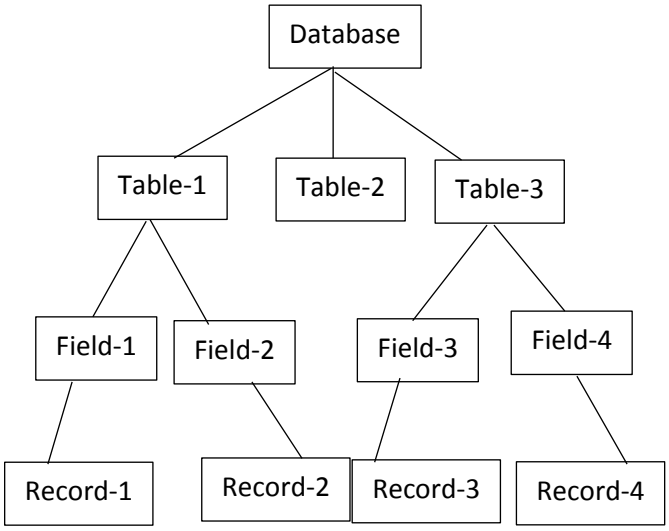


Fig.1 Shows the hierarchy of Database
Defining what level of granularity to use for locking

There, are some problems i.e. lost update, dirty read, problem of summarization and problem of maintain statistics, which you will probable to face while using the Concurrency Control method [3]. It can be distinct as hierarchically breaking up the database into chunks which can be locked. The Multiple Granularity protocol improves concurrency and decreases lock overhead. It keeps the track of what to lock and how to lock. It makes easy to resolve either to lock a data item or to unlock a data item. This type of hierarchy can be graphically represented as a tree [4]. Hierarchical locking is widely used for database indexes [5]. There are so many methods are concurrency control i.e. 2 Phase Locking Protocols [6], time stamp based protocols.

The paper is organized as follows:
Section I contains the Introduction of Concurrent transaction execution and Multiple Granularity, Section II Contains various mode of Intention Locks, Section III contains different types of Locking rules and locks compatibility matrix, Section IV Method- In this section problems are explained and their justification is given with the help of suitable example, Section IV Finally concludes the research work with future direction.

II. INTENTION LOCK MODE

Intention locks are following types

a. Shared Lock (S)
b. Exclusive Lock (X)
c. Intention Shared (IS) Lock – At least one dependent has a shared lock.
d. Intention Exclusive (IX) Lock – At least one dependent has an exclusive lock.
e. Shared and Intention Exclusive (SIX) lock – Node is locked in shared mode and at least one dependent has an exclusive lock (X).

III. RULES OF LOCKING

The locking rules are given below

- Locks are acquired from root to leaf order
- Locks are released from leaf to root order
- Each node has only one lock
- Lock must restrictive

Lock Compatibility Matrix

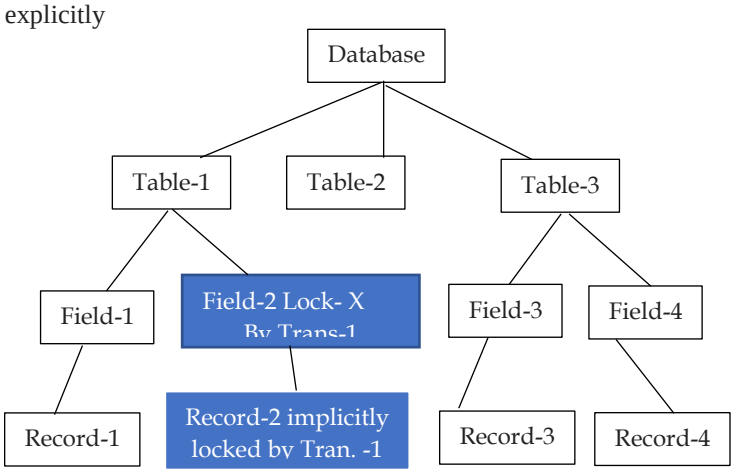
Locks	IS	IX	S	SIX	X
IS	Y	Y	Y	Y	N
IX	Y	Y	N	N	N
S	Y	N	Y	N	N
SIX	Y	N	N	N	N
X	N	N	N	N	N

Table-1

SIX Lock = S + IX. It is new mode of lock which is a safer lock.

IV. METHOD

SCENARIO-1: Suppose transaction-1 wants to lock field-2 explicitly



When transaction-1 locks field-2 explicitly, then record-2 will also lock implicitly by tran. -1 and at this time other transaction cannot access the record-2

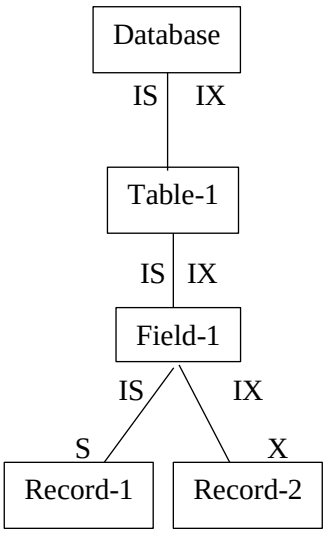
SCENARIO-2: Transaction-2 wants to lock record-2. When transaction-2 tries to lock record-2, permission will not be granted because record-2 already implicitly locked by transaction-1.

SCENARIO-3: Transaction-3 wants to lock table-1 explicitly. When transaction-3 tries to lock the table-1, permission will not be granted. If transaction-3 locks table-1 explicitly then fiels-2 and record-2 will also lock implicitly. But field-2 is already locked by tran. -1.

Example:

1. Transaction-1 wants to read record-1
2. Transaction-2 wants to write record-2
3. Transaction-3 wants to read field-1
4. Transaction-4 wants to read entire database

To find out the combination of transactions will execute concurrently



Solution

Transaction-1 wants to read record-1:

For read the record transaction-1 shall be required shared lock on record-1. When Transaction-1 locks record-1 as Shared mode, Field-1, Table-1 and Database should be locked on Intention Shared (IS) mode

Transaction-2 wants to write record-2:

For write the record transaction-2 shall be required exclusive lock on record-2. When Transaction -2 locks record -2 as exclusive mode, Field-1, Table-1 and Database should be locked on Intention Exclusive (IX) mode. The Intention Shared (IS) lock is compatible with Intention Exclusive (IX) lock as per the table-1. Therefore, Transaction-1 and Transaction-2 will go concurrently.

Transaction-3 wants to read all records of field-1:

For read the record transaction-3 shall be required shared lock on field-1. Shared lock is compatible with Intention Shared (IS) Lock. But Shared Lock (S) is not compatible with Intention Exclusive (IX) Lock as per table-1. Field-1 already has Intention Exclusive (IX) Lock of transaction-2. Hence, Transaction-3 will not execute concurrently.

Transaction-4 wants to read entire database:

For read the database transaction-4 shall be required shared lock on Database. Shared lock is compatible with Intention Shared (IS) Lock. But Shared Lock (S) is not compatible with Intention Exclusive (IX) Lock. Since database already has Intention Exclusive (IX) Lock of transaction-2. Hence, Transaction-4 will not execute concurrently.

V. CONCLUSION AND FUTURE SCOPE

It is found that if entire table needs to lock, then the entire records and tuples of that table will also lock automatically. If you want to access the record then field, table, database should be locked. Therefore, lock is essentially applied multiple granularity. The lock may be applied to only an attribute or to a table or to record or to entire database. It is noted that if you locked parent node explicitly by a transaction then child nodes implicitly locked by that transaction automatically. If that time another transaction wants to lock that particular child node, permission may be denied or granted depending on compatibility among locks.

The limitation of this paper is that I have studied only single method of concurrency control i.e. Multiple Granularity of Locks in multiuser database environment. In future we may work on another method i.e. timestamp based protocol or we may work on comparative study among different methods of concurrency control.

REFERENCE

[1]. Philip A. Bernstein and Eric Newcomer, "Locking", Chapter 6, p.p. 6-1 to 6.35, 2001

[2]. GeeksforGeetks, a computer science Portal for geeks, https://www.geeksforgeeks.org /dbms-concurrency-control-protocol-multiple-granularity-locking/

[3]. Dr. Anil Kumar Singh, "A study of Concurrent transaction execution and their problems in Distributed Database System", International Journal of Computer Sciences and Engineering Vol.6(10), p.p.767-769, E-ISSN: 2347-2693, Oct 2018

[4]. JavaTpoint, Multiple Granularity, https://www.javatpoint.com/dbms-multiple-granularity

[5]. Goetz Graefe, "Hierarchical locking in B-tree indexes", HP Labs 1 Goetz.Graefe@HP.com p.p. 18-42.

[6]. Dr. Anil Kumar Singh, "A study of Preventing Concurrency's Problems using 2-Phase Locking Protocols (2-PL)", International Journal of Computer Sciences and Engineering Vol.6(11), p.p. 39-42, E-ISSN: 2347-2693, November 2018

AUTHOR'S PROFILE

Dr. Anil Kumar Singh is Postgraduate in Computer Science, M.Sc., MCA, PGDCA (1st Position in CSJM University, Kanpur) and Doctorate in Information Technology and having a large 18 years' experience in Academic and Research. Dr. Singh has presented and published more than 20 papers in various National and International Journals and Conferences. His area of expertise is in Computer Network, Database Management, RDBMS, Wi-Fi Technology, Cyber Security, Client/Server Computing, Linux, CISCO and Ethical Hacking. He is a professional life member of Indian Science Congress Association etc. Dr. Singh has a vast experience in academic field and served as Head Computer Center in Dr. GHS-IMR, Kanpur for more than 4 years and presently working as Associate Professor and Academic Head in JIM, Kanpur since year 2005. Dr. Singh has participated various workshops and Short Term Courses organised by the prestigious institutes like I.I.T., Kanpur, I.I.T., Delhi and various technical universities.

Moreover, he has organised various workshops related with Computer Networking, Security and Ethical hacking. Dr. Singh Chaired in the Technical Sessions of International Conferences like IEEE, ICSPICC2016, organized by SSBT College of Engineering, Jalgaon, Maharashtra, IEEE, 2nd ICCIT, organized by Siddhant College of Engineering, Pune, Maharashtra.

